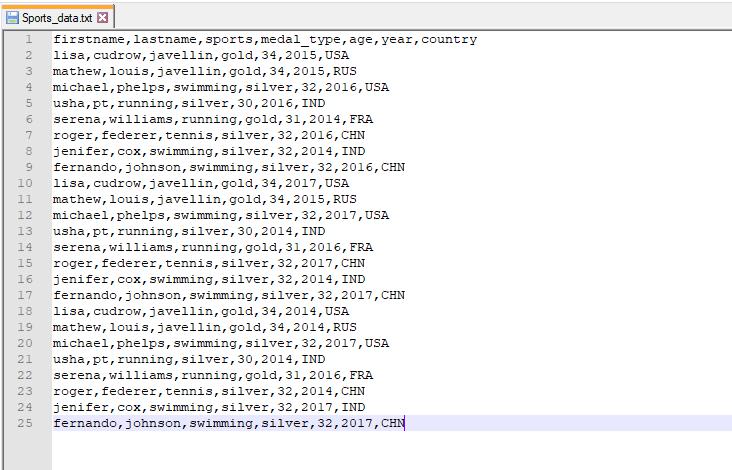
ASSIGNMENT 19.2

**Input Dataset:**

We have sports related data with us:

**Sports\_data**

firstname, lastname, sports, medal\_type, age, year, country



**Problem Statement:**

Using UDFs on data frame:

1. Change firstname, lastname columns into Mr.first\_two\_letters\_of\_firstname<space>lastname for example - michael, phelps becomes Mr.mi phelps

**Solution:**

1. Here is the Spark code snippet to create UDF and its application as per the problem description:

// import required Spark packages

import org.apache.spark.sql.SparkSession

object Assignment19\_2 {

def main(args: Array[String]): Unit = {

* create a SparkSession object that can be used to create various contexts of Spark such as sqlContext val spark = SparkSession

.builder()

.config("spark.sql.warehouse.dir", "file:///c:/tmp/spark-warehouse")

.master("local[\*]")

.getOrCreate()

* initialize sqlContext

val sqlContext = spark.sqlContext

// load input data file – Sports\_data.txt

val sports\_DF = sqlContext.read.option("header", "true").csv("E:\\Acadgild\\Session 19\\ Sports\_data.txt")

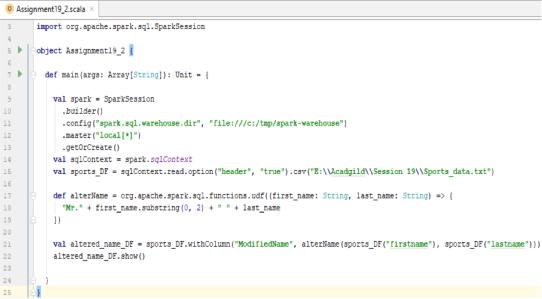
* User Defined Function (UDF) to modify naming convention in the dataset as per instructions def alterName = org.apache.spark.sql.functions.udf((first\_name: String, last\_name: String) => {

"Mr." + first\_name.substring(0, 2) + " " + last\_name })

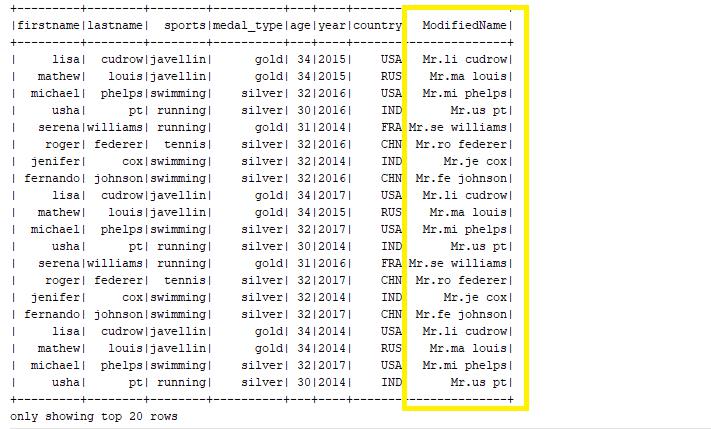
* call the UDF, pass first name and last name as arguments, create new column ‘ModifiedName’
* which will hold the new modified name

val altered\_name\_DF = sports\_DF.withColumn("ModifiedName", alterName(sports\_DF("firstname"), sports\_DF("lastname")))

* show the dataset with new column and its associated values on the console altered\_name\_DF.show()



**Output:**



1. Add a new column called ranking using UDFs on data frame, where: gold medalist, with age >= 32 are ranked as pro

gold medalists, with age <= 31 are ranked amateur silver medalist, with age >= 32 are ranked as expert silver medalists, with age <= 31 are ranked rookie

**Spark code in Scala:**

// import required Spark packages

import org.apache.spark.sql.SparkSession

object Assignment19\_2 {

def main(args: Array[String]): Unit = {

* create a SparkSession object that can be used to create various contexts of Spark such as sqlContext val spark = SparkSession

.builder()

.config("spark.sql.warehouse.dir", "file:///c:/tmp/spark-warehouse")

.master("local[\*]")

.getOrCreate()

* initialize sqlContext

val sqlContext = spark.sqlContext

// load input data file – Sports\_data.txt

val sports\_DF = sqlContext.read.option("header", "true").csv("E:\\Acadgild\\Session 19\\ Sports\_data.txt")

* User Defined Function (UDF) to find ranking of a medalist based on his/her age and medal won def computeRanking = org.apache.spark.sql.functions.udf((medal\_type: String, age: Int) => {

if(medal\_type == "gold" && age >= 32 )

"pro"

else if(medal\_type == "gold" && age <= 31)

"amateur"

else if(medal\_type == "silver" && age >= 32)

"expert"

else if(medal\_type == "silver" && age <= 31)

"rookie"

})

* call the UDF, pass medal type and age as arguments, create new column ‘ranking’ that stores
* value returned by UDF for each record

val ranking\_DF = sports\_DF.withColumn("ranking", computeRanking (sports\_DF ("medal\_type"), sports\_DF("age")))

* show the dataset with new column and its associated values on the console ranking\_DF.show()



**Output:**

